

Serial No. 09/981,453
Response to office action mailed December 14, 2006

Filed On: October 18, 2001

Amendments to the Claims:

The listing of Claims will replace all prior versions and listings of the Claims in the application:

Listing of Claims

1. – 20. (Canceled)

21. (Currently Amended) A method of operating a business services application for retrieving data with delivery technologies, the method comprising:

developing custom application code in a subclass of a BusinessService class, the custom application code responsive to a request for data initiated by the delivery technologies;

translating the request to a first document object model document with an ApiService class;

~~selectively during the translation, limiting the data structure of the first document object model document with a Message class and a Field class during the translation to representation as an input message with a plurality of fields, wherein units of data included in each of the fields is limited to a data type that is pre-specified in the business services application;~~

executing the custom application code to retrieve data based on the first document object model document;

reading data into a second document object model document with the ApiService class;

while the data is read in, ~~selectively limiting the data structure of the second document object model document with the Message class and the Field class to representation as an output message with a plurality of fields, wherein units of data included in each of the fields is limited to a data type that is pre-specified in the business services application;~~ and

translating the second document object model document with the ApiService class based on the delivery technology.

Serial No. 09/981,453

Filed On: October 18, 2001

Response to office action mailed December 14, 2006

22. (Currently Amended) The method of claim 21, wherein ~~selectively~~ limiting the data structure of the first document object model document comprises ~~setting~~ populating a plurality of text nodes within the first document object model document ~~to a unit of data identified by a tag in the request with request parameters contained in the request that are translated to a format identified with the pre-specified data type.~~

23. (Canceled)

24. (Currently Amended) The method of claim 22~~3~~, wherein ~~selectively~~ limiting the data structure of the first document object model document further comprises limiting the predetermined datatype to a format of a string datatype.

25. (Currently Amended) The method of claim 21, wherein ~~selectively~~ limiting the data structure of the first document object model document comprises ~~setting~~ populating an attribute node within the first document object model document ~~to with~~ an attribute identified by a request name parameter in the request. of the request that is translated to a format identified with the pre-specified data type.

26. (Currently Amended) The method of claim 21, further comprising selecting, as a function of a mode debug flag, to use one of a short field name ~~and~~ or a long field name as a field name for each of ~~the a plurality of~~ fields in the first and second document object model documents.

27. (Currently Amended) The method of claim 21, wherein the pre-specified data types are selected from a pre-specified group of data types consisting of a string datatype, a long datatype, an integer data type, a boolean data type and a group data type ~~translating the request comprises representing an input message with the first document object model document.~~

Serial No. 09/981,453
Response to office action mailed December 14, 2006

Filed On: October 18, 2001

28. (Currently Amended) The method of claim 21, wherein limiting the data structure of the first document object model document comprises loading a static declaration of a datatype based on a list of fields expected in the request reading data into a second document object model ~~comprises representing an output message with the second document object model document.~~

29. (Currently Amended) The method of claim 21, wherein selectively limiting the data structure of the second document object model document comprises setting, based on a datatype populating a plurality of text nodes within the second document object model document to with data read in to the second document object model document, wherein the format of the data that is read in is converted based on the data type.

30. (Currently Amended) The method of claim 21, wherein selectively limiting the data structure of the second document object model document comprises setting, as a function of a datatype populating an attribute node within the second document object model document to with an attribute read in to the second document object model document with the data that is translated to a format identified with the pre-specified data type.

31. (Canceled)

32. (Previously Presented) The method of claim 21, wherein translating the second document object model comprises translating the second document object model document to extensible markup language text.

33. (Previously Presented) The method of claim 21, wherein translating the second document object model comprises translating the second document object model document to at least one of a hypertext markup language and a website meta language as a function of at least one extensible stylesheet language stylesheet.

Serial No. 09/981,453
Response to office action mailed December 14, 2006

Filed On: October 18, 2001

34. – 40. (Canceled)

41. (Currently Amended) A system for leveraging extensible markup language technology to provide an interface between a back-end systems layer and a front-end systems layer, the system comprising:

a server computer;

an ApiService class operable within the server computer to direct the translation of a request to an input message that includes a plurality of fields;

a document object model class operable within the server computer to represent the input message as a document object model document;

a Message class and a Field class operable within the server computer as wrapper of the document object model class to restrict manipulation and standardize the content of the document object model document;

a MESSAGEDEFINITION class operable in the server, wherein the MESSAGEDEFINITION class includes a listing of pre-specified fields each of which describe a corresponding pre-specified data type, and wherein the Message class and the Field class are further operable within the server during translation to limit a format of corresponding fields included in the input message to a predetermined data structure based on the described corresponding pre-specified data type; and

a BusinessService class operable within the server computer to direct the execution of custom application code as a function of the input message, wherein the custom application code includes a pre-specified data type to limit the format of those fields included in the input message that do not correspond to the listing of pre-specified fields.

Serial No. 09/981,453
Response to office action mailed December 14, 2006

Filed On: October 18, 2001

42. (Currently Amended) The system of claim 41, wherein the custom application code is operable to process the input message to retrieve data, the data translatable with the document object model class, the Message class and the Field class to an output message in the form of a document object model document with restricted manipulation and standardized content based on the pre-specified data type included in the custom application code that, during translation, is operable to limit a format of each of a plurality of fields included in the output message to a predetermined data structure.
43. (Previously Presented) The system of claim 42, wherein the ApiService class is operable to direct the conversion of the output message to a presentation format defined by the request.
44. (Previously Presented) The system of claim 41, wherein the input message and the output message comprises a root element and a plurality of sub-elements.
45. (Previously Presented) The system of claim 41, further comprising a Fldtypes class operable within the server computer, wherein the Fldtypes class comprises definitions of the format of datatypes for fields within the input message.
46. (Currently Amended) The system of claim 41, wherein the document object model document comprises a plurality of field names, the field names selectable with a mode debug flag as one of a first short field name and a second long field name.
47. (Currently Amended) The system of claim 46, wherein the first short field name and the second long field name are defined in the a MESSAGEDEFINITION class operable within the server computer.
48. (Previously Presented) The system of claim 41, wherein the document object model class comprises a Document class, a document object model Element class and a plurality of

Serial No. 09/981,453
Response to office action mailed December 14, 2006

Filed On: October 18, 2001

ProcessingInstruction classes, the Message class operable as a wrapper of the Document class, the document object model Element class and the Processing Instruction classes.

49. (Previously Presented) The system of claim 41, wherein the document object model class comprises a document object model setAttribute method, Field class operable as a wrapper of the document object model setAttribute method.

50. (Previously Presented) The system of claim 41, wherein the BusinessService class comprises a subclass of custom application code responsive to the request.

51. (Previously Presented) A method of leveraging extensible markup language technology to interface a front-end systems layer with a back-end systems layer, the method comprising:
receiving one of a plurality of predetermined requests initiated with any one of a plurality of delivery technologies;
converting the request to a plurality of fields based on request parameters included in the request;
limiting a datatype of data included in the fields to a predefined group of datatypes;
extracting the request parameters based on the datatype; and
accessing data responsive to the request based on the extracted request parameters.

52. (Previously Presented) The method of claim 51, wherein the datatype of data included in the fields is predefined by the request.

53. (Previously Presented) The method of claim 51, wherein the datatype of data included in the fields is loaded from a static declaration of the datatype included in a MESSAGEDEFINITION class.

Serial No. 09/981,453
Response to office action mailed December 14, 2006

Filed On: October 18, 2001

54. (Previously Presented) The method of claim 51, wherein converting the request comprises translating the request to extensible markup language structure that is limited to the predefined group of datatypes.

55. (Previously Presented) The method of claim 51, further comprising converting the data responsive to the request into a plurality of fields with a datatype that is limited to the predefined group of datatypes based on the request parameters, and translating the fields into a format indicated by the request to be compatible with the one of the delivery technologies that made the request.

56. (Previously Presented) The method of claim 51, wherein converting the request comprises translating the request into a document object model document having a predefined name that is included in the request and a plurality of tags having attributes indicative of a corresponding datatype.

57. (Previously Presented) The method of claim 56, further comprising translating the data responsive to the request into another document object model document to represent an output message with datatypes that are limited to the group of predefined datatypes, and converting the another document object model into a format indicated by the request to be compatible with the one of the delivery technologies that made the request.

58. (Currently Amended) The method of claim 51, wherein limiting the datatype comprises limiting the data to representation as one of integer, long, Boolean, string ~~and~~ or group.

59. (Previously Presented) The method of claim 51, further comprising generating a structure for a response to the request in extensible mark up language that includes the data responsive to

Serial No. 09/981,453
Response to office action mailed December 14, 2006

Filed On: October 18, 2001

the request, wherein, in the response, the data responsive to the request is limited to the predefined group of datatypes.

60. (Currently Amended) The method of claim 51, wherein accessing data responsive to the request comprises limiting the data responsive to the request that is retrieved to representation as one of integer, long, Boolean, string and or group.

61. (Previously Presented) The method of claim 51 further comprising:
converting the data responsive to the request to a plurality of fields based on a datatype of the data responsive to the request;
limiting the datatype of the data responsive to the request included in the fields to one of a predefined group of datatypes; and
providing the data responsive to the request as a response.

62. (Previously Presented) The method of claim 51, wherein extracting the request parameters comprises executing custom application code that is responsive to a request name included in the request.

63. (Previously Presented) The method of claim 62, wherein executing custom application code comprises setting the root element to a message name as a function of the request name parameter.

64. (Currently Amended) An e-commerce architecture for providing a framework to interface delivery technologies with data, the e-commerce architecture comprising:
a server computer operable to execute instructions to convert a request to a first document object model document in an extensible markup language, the first document object model document comprising a plurality of request parameters extracted from the request;

Serial No. 09/981,453
Response to office action mailed December 14, 2006

Filed On: October 18, 2001

the server computer operable to execute instructions to restrict the conversion to the first document object model document to based on a listing of data types that are pre-specified for the request parameters, wherein the data types limit the data structure of a plurality of fields included in the first document object model document to a predetermined data structure specified by the data types ~~standardize the content and limit a manipulative capability of the extensible markup language within a document object model class;~~

the server computer operable to execute instructions to retrieve data responsive to the request and convert the data to a second document object model document in the extensible markup language based on the request parameters; and

the server computer operable to execute instructions to restrict the conversion of the data to the second document object model document to limit the data structure of a plurality of fields included in the second document object model document to a predetermined data structure specified by the data types ~~similarly standardize the content and limit the manipulative capability of the extensible markup language within the document object model class.~~

65. (Previously Presented) The e-commerce architecture of claim 64, wherein the instructions to restrict the conversion of the first and second document object model documents further comprise instructions executable by the server computer to identify the first and second document object model documents with a predefined name included in the request.

66. (Previously Presented) The e-commerce architecture of claim 64, wherein the instructions to restrict the conversion of the first and second document object model documents further comprise instructions executable by the server computer to create a plurality of element nodes and populate a plurality of corresponding text nodes with the respective request parameters and the respective data.

Serial No. 09/981,453
Response to office action mailed December 14, 2006

Filed On: October 18, 2001

67. (Currently Amended) The e-commerce architecture of claim 66, wherein the instructions to restrict the conversion of the first and second document object model documents further comprise instructions executable by the server computer to define the a data type of each of the text nodes from among a predefined group of data types.

68. (Previously Presented) The e-commerce architecture of claim 64, wherein the instructions to restrict the conversion comprises a Message class operable as a wrapper of a plurality of classes within the document object model class that include a document class and a document object model element class.

69. (Previously Presented) The e-commerce architecture of claim 64, wherein the instructions to restrict the conversion comprises a Field class operable as a wrapper of a document object model setAttribute method in a document object model element class.

70. (Previously Presented) The e-commerce architecture of claim 64, wherein the instructions to retrieve data responsive to the request are identified with a request name that is included in the request.

71. (Currently Amended) The method of claim 21, wherein ~~selectively~~ limiting the data structure of the first document object model comprises standardizing the format of the document object model to be substantially similar for a similar request received from any one of the delivery technologies.

72. (Currently Amended) The method of claim 71, wherein ~~selectively~~ limiting the data structure of the second document object model comprises standardizing the format of the second document object model to be compatible with any one of the delivery technologies.

Serial No. 09/981,453
Response to office action mailed December 14, 2006

Filed On: October 18, 2001

73. (Previously Presented) The method of claim 72, wherein executing the custom application code comprises executing the same custom application code for a similar request from any one of the delivery technologies to provide a response.

74. (Previously Presented) The method of claim 21, wherein executing the custom application code comprises executing the same custom application code for a similar request from any one of the delivery technologies.

75. (Currently Amended) The method of claim 74, wherein while the data is read in, selectively limiting the data structure of the second document object model document comprises similarly limiting the second document object model in response to similar requests from any of the delivery technologies.

76. (Previously Presented) The system of claim 41, wherein the Message class and the Field class are operable during representation of the input message as the document object model document to restrict manipulation of the document object model document.

77. (Previously Presented) The system of claim 41, wherein the Message class is operable to restrict creation of the element nodes and population of the corresponding text nodes and the Field class is operable to restrict the data types of text and attribute nodes included in the first document object model document.

78. (Canceled)

Serial No. 09/981,453
Response to office action mailed December 14, 2006

Filed On: October 18, 2001

79. (Currently Amended) The method of claim 4178, wherein the ~~predetermined group of~~ pre-specified datatypes are selected from the group consisting of integer, long, Boolean, string and group.